



www.ijarcsse.com

Volume 2, Issue 5, May 2012

ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Independent Task Scheduling in Cloud Computing by Improved Genetic Algorithm

Pardeep Kumar*, Amandeep Verma

UIET, Panjab University, Chandigarh, India

pardeep_prof@yahoo.com

Abstract— Scheduling is a critical problem in Cloud computing, because a cloud provider has to serve many users in Cloud computing system. So scheduling is the major issue in establishing Cloud computing systems. A good scheduling technique also helps in proper and efficient utilization of the resources. Many scheduling techniques have been developed by the researchers like GA (Genetic Algorithm), PSO (Particle Swarm Optimization), Min-Min, Max-Min, X-Sufferage etc. This paper proposes a new scheduling algorithm which is an improved version of Genetic Algorithm. In the proposed scheduling algorithm the Min-Min and Max-Min scheduling methods are merged in standard Genetic Algorithm. Min-Min, Max-Min and Genetic Scheduling techniques are discussed and in the last the performance of the standard Genetic Algorithm and proposed improved Genetic Algorithm is compared and is shown by graphs.

Keywords— Cloud, Cloud Computing, Min-Min, Max-Min, Genetic Algorithm, Improved Genetic Algorithm.

I. INTRODUCTION

As the IT technologies are growing day by day, the need of computing and storage resources are rapidly increasing. To invest more and more equipments is not an economic method for an organization to satisfy the even growing computational and storage need. So Cloud Computing has become a widely accepted paradigm for high performance computing, because in Cloud Computing all type of IT facilities are provided to the users as a service. Cloud computing is a category of sophisticated on-demand computing services initially offered by commercial providers, such as Amazon, Google, and Microsoft.

In Cloud Computing the term Cloud is used for the service provider, which holds all types of resources for storage, computing etc. Mainly three types of services are provided by the cloud. First is Infrastructure as a Service (IaaS), which provides cloud users the infrastructure for various purposes like the storage system and computation resources. Second is Platform as a Service (PaaS), which provides the platform to the clients so that they can make their applications on this platform. Third is Software as a Service (SaaS), which provides the software to the users; so users don't need to install the software on their own machines and they can use the software directly from the cloud.

Due to the wide range of facilities provided by the cloud computing, the Cloud Computing is becoming the need of the IT industries. The services of the Cloud are provided through the Internet. The devices that want to access the services of the Cloud should have the Internet accessing capability. Devices need to have very less memory, a very light operating

system and browser. Cloud Computing provides many benefits: it results in cost savings because there is no need of initial installation of much resource; it provides scalability and flexibility, the users can increase or decrease the number of services as per requirement; maintenance cost is very less because all the resources are managed by the Cloud providers.

The rest of this paper is organized as follows. Section II briefly describes related research for scheduling in grid computing and Cloud Computing. Section III discusses about the scheduling techniques. Section IV describes the Genetic Algorithm. Section V gives the idea about the new Improved Genetic Algorithm telling how we can combine Min-Min and Max-Min in genetic Algorithm. Section VI is having the simulations and results. Section VII tells about the future scope and conclusion of this paper.

II. RELATED WORK

Scheduling of tasks is a critical issue in Cloud Computing, so a lot of researches have been done in this area. The basic ideas about scheduling in Cloud Computing and scheduling techniques are discussed in [1] and [2]. The Scheduling using Genetic Algorithm and other modified versions of Genetic Algorithms are discussed in [3] up to [9]. We have discussed in this paper three scheduling techniques, which are Min-Min, Max-Min and Genetic Algorithm.

III. SCHEDULING TECHNIQUES

There are various scheduling techniques, but we are discussing here three of them using which we have proposed

the improved Genetic Algorithm. The scheduling techniques Min-Min and Max-Min are discussed first and their performance is shown against the sample data of TABLE I.

A. Min-Min Algorithm

Min-Min begins with a set of tasks which are all unassigned. First, it computes minimum completion time for all tasks on all resources. Then among these minimum times the minimum value is selected which is the minimum time among all the tasks on any resources. Then that task is scheduled on the resource on which it takes the minimum time and the available time of that resource is updated for all the other tasks. It is updated in this manner; suppose a task is assigned to a machine and it takes 20 seconds on the assigned machine, then the execution times of all the other tasks on this assigned machine will be increased by 20 seconds. After this the assigned task is not considered and the same process is repeated until all the tasks are assigned resources.

B. Max-Min Algorithm

Max-Min is almost same as the min-min algorithm except the following: in this after finding out the completion time, the minimum execution times are found out for each and every task. Then among these minimum times the maximum value is selected which is the maximum time among all the tasks on any resources. Then that task is scheduled on the resource on which it takes the minimum time and the available time of that resource is updated for all the other tasks. The updating is done in the same manner as for the Min-Min. All the tasks are assigned resources by this procedure.

We have implemented the logic of Min-Min and Max-Min algorithms on the execution time values as given in the TABLE I below. We have assumed four machines and six tasks.

TABLE I
EXECUTION TIMES

	M0	M1	M2	M3
T0	200	250	220	300
T1	150	170	190	160
T2	300	320	180	360
T3	400	380	350	310
T4	100	120	140	160
T5	220	250	280	200

By using different scheduling techniques, the tasks are assigned in a different sequence to different machines for execution. When we apply the different scheduling techniques Min-Min and Max-Min, then the tasks will be assigned to the machines as given in the following figures:

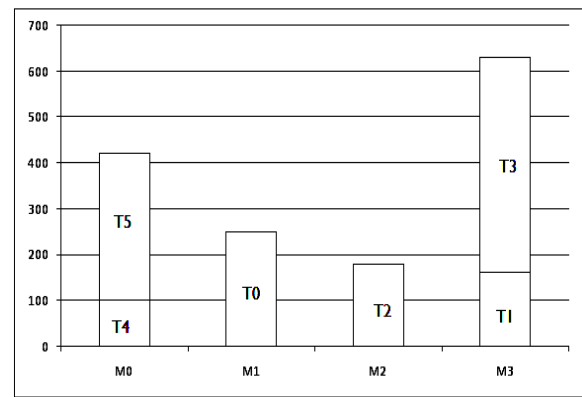


Fig. 1 Task assignment by Min-Min algorithm

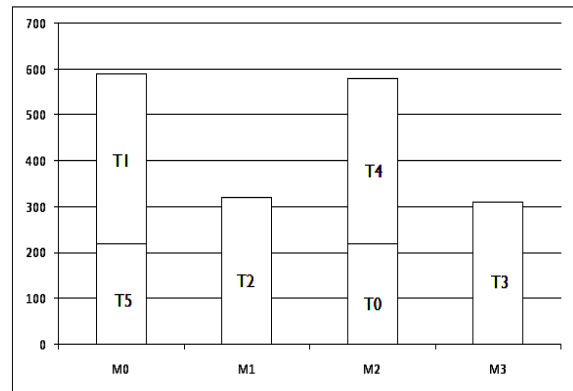


Fig. 2 Task assignment by Max-Min algorithm

There is a term “makespan” in Min-Min and Max-Min scheduling techniques, which is the maximum execution time on any machine among the machines on which the tasks are scheduled. For example, in Fig. 1, “630” is the makespan because it is the maximum execution time among the four machines.

In Fig. 1 and Fig. 2, the x-axis represents the different machines and y-axis represents the execution times. We have got the following different values of makespans by the two techniques:

Method used	Makespan
Min-Min	630
Max-Min	590

Based on the different execution times of tasks on resources, one technique can outperforms the other and the assignment of resources to the tasks can change i.e. if any task is assigned to a machine if we use one technique; the same task can be assigned to another machine if we use other technique.

IV. GENETIC ALGORITHM

Genetic algorithm is a method of scheduling in which the tasks are assigned resources according to individual solutions (which are called schedules in context of scheduling), which tells about which resource is to be assigned to which task. Genetic Algorithm is based on the biological concept of population generation. The main terms used in genetic algorithms are:

A. Initial Population

Initial population is the set of all the individuals that are used in the genetic algorithm to find out the optimal solution. Every solution in the population is called as an individual. And every individual is represented as a chromosome for making it suitable for the genetic operations. From the initial population the individuals are selected and some operations are applied on those to form the next generation. The mating chromosomes are selected based on some specific criteria.

B. Fitness Function

A fitness function is used to measure the quality of the individuals in the population according to the given optimization objective. The fitness function can be different for different cases. In some cases the fitness function can be based on deadline, while in cases it can be based on budget constraints.

C. Selection

We use the proportion selection operator to determine the probability of various individuals genetic to the next generation in population. The proportional selection operator means the probability which is selected and genetic to next generation groups is proportional to the size of the individual's fitness.

D. Crossover

We use single-point crossover operator. Single-point crossover means only one intersection was set up in the individual code, at that point part of the pair of individual chromosomes is exchanged.

E. Mutation

Mutation: - Mutation means that the values of some gene locus in the chromosome coding series were replaced by the other gene values in order to generate a new individual. Mutation is that negates the value at the mutate points with regard to binary coded individuals.

Genetic Algorithm works in the following manner:

1. Begin
2. Initialize population with random candidate solutions
3. Evaluate each candidate
4. Repeat Until (termination condition is satisfied)
 - a. Select parents
 - b. Recombine pairs of parents
 - c. Mutate the resulting offsprings
 - d. Evaluate new candidate
 - e. Select individuals for the next generation;
5. End

V. IMPROVED GENETIC ALGORITHM

In Genetic Algorithm the initial population is generated randomly, so the different schedules are not so much fit, so when these schedules are further mutated with each other, there are very much less chances that they will produce better child than themselves. We have provided an idea for generating initial population by using the Min-Min and Max-

Min techniques for Genetic Algorithms. As discussed in Genetic Algorithm; the solutions that are fit, give the better generations further when we apply genetic operators on them and hence if Min-Min and Max-Min will be used for the individual generation, we will get the better initial population and further the better solutions than in the case of standard Genetic Algorithm in which initial population is chosen randomly.

The new Improved Genetic Algorithm is like given below:

1. Begin
2. Find out the solution by Min-Min and Max-Min
3. Initialize population by the result of Step 2
4. Evaluate each candidate
5. Repeat Until (termination condition occur)
 - a. Select parents
 - b. Recombine pairs of parents
 - c. Mutate the resulting offsprings
 - d. Evaluate new candidate
 - e. Select individuals for next generation
6. End

VI. SIMULATIONS AND RESULTS

We have used CloudSim as a simulator for checking the performance of our improved algorithm and the standard Genetic Algorithm. CloudSim is an extensible simulation toolkit that enables modeling and simulation of Cloud computing systems and application provisioning environments. The CloudSim toolkit supports both system and behavior modeling of Cloud system components such as data centers, virtual machines (VMs) and resource provisioning policies. It implements generic application provisioning techniques that can be extended with ease and limited efforts.

We have considered Virtual Machines as resource and Cloudlets as tasks/jobs. We have checked the performance of the algorithms in two cases: in first case, we have fixed the number of virtual machines and varied the number of cloudlets; in second case we case we have fixed the number of cloudlets and varied the number of virtual machines. The makespans that the algorithms produce are shown in tables and the graphs corresponding to these tables have been shown.

In first case, we have fixed the number of virtual machines as 10 and we are varying the number of cloudlets from 10 to 40 with a difference of 10. We have run each algorithm 10 times and the average of these 10 runs is noted down in Table II shown below:

TABLE II. MAKESANS FOR FIXED VMs AND VARYING CLOUDLETS

VMs fixed : 10		Cloudlets varying			
		10	20	30	40
Method Used	Improved Genetic	8	26.1	60.9	113.5
	Standard Genetic	12.4	44.7	86.5	146.8

The performance of the first case according to the noted values is shown in graph of Fig. 3, in which x-axis shows the

number of cloudlets and the y-axis shows the makespans and the number of virtual machines is fixed as 10:

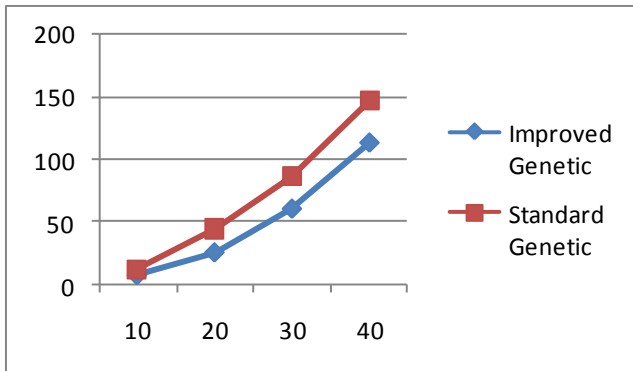


Fig. 3. Graph for makespans for fixed VMs and varying Cloudlets

In second case, we have fixed the number of cloudlets 40 and we are varying the number of virtual machines from 10 to 40 with a difference of 10. We have run each algorithm 10 times and the average of these 10 runs is noted down in the Table III shown below:

TABLE III. MAKESANS FOR FIXED CLOUDLETS AND VARYING VMs

Cloudlets fixed : 40		VMs varying			
		10	20	30	40
Method Used	Improved Genetic	113.5	33.2	17.1	10
	Standard Genetic	146.8	84.7	54	44.8

The performance of the first case according to the noted values is shown in graph of Fig. 4, in which x-axis shows the number of virtual machines and the y-axis shows the makespans and the number of cloudlets is fixed as 40:

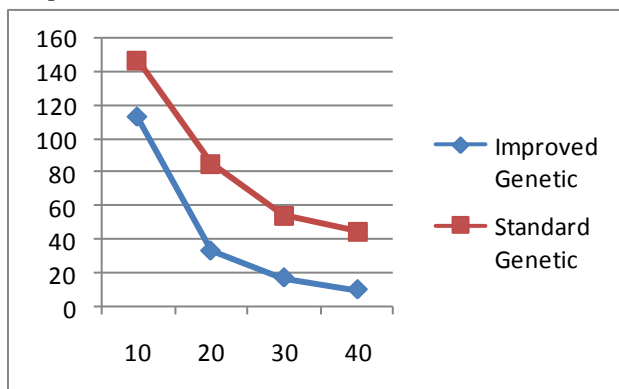


Fig. 4. Graph for makespans for fixed Cloudlets and varying VMs

From both the graphs, it can be observed that the makespan of the Improved Genetic Algorithm is less than that of Standard Genetic Algorithm. So the new improved Genetic Algorithm can help in reducing overall execution time of the tasks and in proper utilization of resources.

VII. CONCLUSION AND FUTURE SCOPE

We have designed and tested an algorithm which is made by combining Min-Min and Max-Min in Genetic Algorithm. It is able to schedule multiple jobs on multiple machines in an efficient manner such that the jobs take the minimum time for completion. This technique can be adapted in the cloud computing systems for better scheduling of tasks to resources, so that the users' tasks can be completed in as minimum time as possible.

REFERENCES

- [1] Huang Q.Y., Huang T.L., "An Optimistic Job Scheduling Strategy based on QoS for Cloud Computing", IEEE International Conference on Intelligent Computing and Integrated Systems (ICISS), 2010, Guilin, pp. 673-675, 2010
- [2] Hsu C.H. and Chen T.L., "Adaptive Scheduling based on Quality of Service in Heterogeneous Environments", IEEE 4th International Conference on Multimedia and Ubiquitous Engineering (MUE), 2010, Cebu, pp. 1-6, 2010
- [3] Yin H., Wu H., Zhou J., "An Improved Genetic Algorithm with Limited Iteration for Grid Scheduling", IEEE Sixth International Conference on Grid and Cooperative Computing, 2007. GCC 2007, Los Alamitos, CA, pp. 221-227, 2007
- [4] Guo G., Ting-lei H., Shuai G., "Genetic Simulated Annealing Algorithm for Task Scheduling based on Cloud Computing Environment", IEEE International Conference on Intelligent Computing and Integrated Systems (ICISS), 2010, Guilin, pp. 60-63, 2010
- [5] Wang J., Duan Q., Jiang Y., Zhu X., "A New Algorithm for Grid Independent Task Schedule: Genetic Simulated Annealing", IEEE World Automation Congress (WAC), 2010, Kobe, pp. 165-171, 2010
- [6] Verma R., Dhingra S., "Genetic Algorithm for Multiprocessor Task Scheduling", IJCSMS International Journal of Computer Science and Management Studies, Vol.1, Issue 02, pp. 181-185, 2011
- [7] Loukopoulos T., Lampsas P., Sigalas P., "Improved Genetic Algorithms and List Scheduling Techniques for Independent Task Scheduling in Distributed Systems", IEEE, Eighth International Conference on Parallel and Distributed Computing, Applications and Technologies, Adelaide, SA, pp. 67-74, 2007
- [8] Zhao C., Zhang S., Liu Q., Xie J., Hu J., "Independent Tasks Scheduling Based on Genetic Algorithm in Cloud Computing", IEEE 5th International Conference on Wireless Communications, Networking and Mobile Computing WiCom '09, Beijing, pp.1-4, 2009
- [9] Abdulal W., Jadaan O.A., Jabas A., Ramachandram S., "Genetic Algorithm for Grid Scheduling using Best Rank Power", IEEE World Congress on Nature & Biologically Inspired Computing, 2009. NaBIC 2009, Coimbatore, pp. 181-186, 2009